

# Data-Aware Picking for Medical Models

Eva Monclús<sup>1</sup> and Pere-Pau Vázquez<sup>1</sup> and Isabel Navazo<sup>1</sup>

ViRVIG-LSI, Universitat Politècnica de Catalunya, c. Jordi Girona 1–3, Edif. Omega,  
08034 Barcelona, Spain  
{emonclus, ppau, isabel}@lsi.upc.edu

**Abstract.** Medical doctors are often faced with the problem of selecting anchoring points in 3D space. These points are commonly used for measurement tasks, such as lengths of bones, or dimensions of pathological structures (e. g. tumours). Since previous research indicates that measurement tasks can be usually carried out more efficiently in VR environments than in desktop-based systems, we have concentrated on the development of selection tools for medical models. These models have a set of particularities such as the presence of semi-transparencies, and there is a lack of tools for measurement support for such models in VR environments. Our VR-based interaction technique uses the data to automatically generate candidate anchor points, and it is specially focused on the efficient selection of 3D points in datasets rendered using methods with semi-transparency such as Direct Volume Rendering. We will show that our method is effective, precise, and reduces the time and amount of movements required to set the anchor points as compared with other classical techniques based on clipping planes. We also provide a couple of improvements tailored to reduce the inherent imprecision of 3D devices due to hand vibration, and the flexibility in transfer function selection for anchor point definition.

## 1 Introduction

In medical applications, the quantitative analysis of spatial relations between structures is crucial for many tasks such as diagnosis, treatment and surgical planning, and documentation. These measurements include, among others, the extension of pathological structures or the distance between pathological structures and structures at risk (blood vessels). In the field of Neurosurgery, for instance, distance between the brain surface and the ventricles is an important parameter that may determine the surgical trajectory. However, the use of these 2D images does not facilitate the perception of the relative position of the structures, and as a consequence, it is often rough for accurately locating anchor points and thus obtaining precise 3D magnitudes.

Advances in volume visualization allow for the 3D reconstruction and analysis of anatomical structures from a stack of intensity-based images acquired from, usually, CT or MRI modalities. Initial algorithms identified and extracted the isosurfaces of the anatomical structures as triangle meshes. This process is time-consuming and loses contextual information. Later methods directly render the

volume (Direct Volume Rendering, or DVR) by assigning color and opacity to the samples as a function of its density by using pre-defined transfer functions. Semi-transparencies provide a means to increase the amount of information visible to the users, and facilitate establishing spatial relationships between elements such as the skin and the bones. This led the development of new techniques for anchor point definition in desktop-based applications. However, occlusions still remain as a problem for the selection. This is often addressed with the introduction of clipping planes showing the volume cut mapped on them.

Reitinger *et al.* [1] found that measurement tasks can be carried out more efficiently in a Virtual Reality environment than in a desktop setup. The cost reduction of VR systems and GPUs is helping the introduction of such systems in surgical planning and diagnose. Stereo vision facilitates the perception of the relative position of anatomical structures, although occlusion remains as a research problem. It is important to note that we are not interested in selecting a concrete structure, but a point on it, without any previous surface extraction nor segmentation process. Additionally, the occlusion problem remains, as well as the fact that a VR selection environment is not familiar to medical experts.

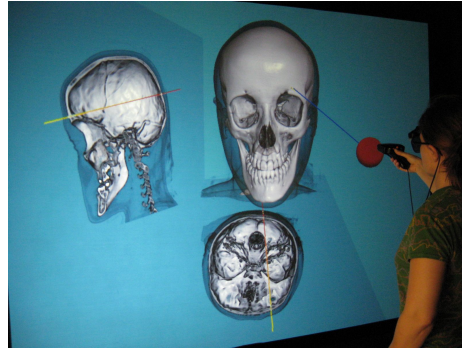
The purpose of this paper is to provide an easy-to-use tool for the fast and accurate selection of 3D points on the implicitly defined surfaces of anatomical structures present in a volume dataset in a virtual environment. This is an extension of our previous contribution [2]. Here we include two techniques to reduce the effects of trembling hands, we include the use of the Virtual Magic Lantern [3] metaphor to increase the flexibility in transfer function usage for point selection, and we provide a broader user study on the benefits of our technique. The contributions of our paper are:

- We review DAAPMed, an anchor point selection tool suitable for medical models in VR environments. The system assists the user by automatically computing data-aware candidate anchor points (see Figure 1).
- A series of visual cues that provide feedback on the ray position through the use of mirror views and supporting planes.
- The reduction of effects of trembling hands in picking.
- A user study analysing the accuracy and performance of the picking method.

The implementation of all these components (both GPU and CPU) guarantees real-time feedback and interaction. This is an important issue in VR environments which require rendering the model twice. We will show that the technique is effective and accurate, and reduces the amount of displacements and time required for the selection as compared with a classical clipping plane technique in a VR environment. We also introduce the use of the Virtual Magic Lantern [3] to modify the transfer function that determines the candidate points.

## 2 Related Work

In a pioneer work, Hinckley *et al.* [4] proposed a 3D user interface for pre-operative neurosurgical planning based on the physical manipulation of familiar



**Fig. 1.** User interacting with a model using the DAAPMed metaphor.

real-world objects (head, cutting-plane and stylus-shaped props) in free space to access and manipulate a virtual model. This approach offers the possibility of selecting anchor points in a brain model. They use a clipping plane to access occluded or interior points in the brain and then select anchor points on it as the intersection of the linear trajectory defined by the stylus and the cutting-plane.

Preim *et al.* [5] introduced a set of applicable tools for the computation of distances, angles, and volumes in 3D visualizations. The tools are 3D virtual objects such as a distance line, a ruler and angular measurements that are manipulated using the mouse in a desktop-platform. They allow to determine anchor points on the surface of the pre-segmented anatomical structures. Rossling *et al.* [6] proposed a method for the automatic determination of different distance-based measures (shortest distance, diameters and wall thickness) also on segmented anatomic structures. The necessity of this kind of tool is justified by the fact that manual distance calculation is tedious and imprecise in single 2D slices, and although it is possible to achieve an accurate result in 3D, it would also be tiresome. However, completely automatic measurements are difficult to generalize due to the great variety of problems and anatomical structures. Notice that both previous approaches [5, 6] work on triangle mesh representations, so a surface extraction process is needed previously to use them. Moreover, they always select the nearest visible point and they do not deal with semi-transparent models. Reitingner *et al.* [1] presented a 3D measurement toolkit developed for liver surgery especially tailored for a VR platform. Their measurements include distance, volume, and angles. Their evaluation indicated that VR-based measurement tools have a sufficient benefit compared to 2D desktop-based systems in terms of task completion time. In terms of accuracy, slightly better results in most of the tasks were achieved. The anatomical structures models (liver, vessels,...) are computed through segmentation from CT scans and they are represented by opaque triangle meshes where the user may select points by using a virtual pencil. Hagerdorn *et al.* [7] proposed a set of tools for performing measurements in a virtual reality visualization environment. A 3D Rubberbanding line for selecting *free* points in the scene is proposed. They use clipping planes

for accessing interior parts of the volume dataset. Their scene is also composed by triangle meshes.

Segmentation and surface extraction are time consuming operations. To overcome this problem, Hastreiter *et al.* [8] suggest direct volume rendering of the entire data volume, giving insights to interior and superimposed information. In order to inspect interior structures, independent clipping planes provide an intuitive way to virtually cut off parts of the volume data set. Then, anchor points can be interactively placed on the clipping planes. Gallo *et al.* [9] present a VR system for the exploration of volume datasets using a Wiimote. Apart from the basic interaction techniques for navigating they propose a mechanism of selection of points based on the classical ray-casting technique adding the mechanism of *fishing reel* in which the users can move the cursor closer or farther away by using two buttons in order to accurately locate a mark. Unfortunately, points' positions are not aware of the isosurfaces and no visual cue is used to reveal the cursor when it is moved into an occluded region.

Many researchers have investigated 3D object selection techniques for general -non medical- VR applications [10]. In this area, *Ray-based* techniques have shown a better performance than *point-based* techniques. These former approaches are usually based on a cone or a ray. Since our interest is on accurate anchor point selection, we only consider ray-based tools. In order to solve the inherent problem of multiple intersection candidates, several disambiguation techniques have been proposed. It is important to note that most of these VR selection metaphors are focused on selection and manipulation of objects (not points) in populated scenarios, and thus they were not specially concerned about accuracy in point selection. Grossman *et al.* [11] explored 3D selection techniques for volumetric displays and proposed new ray cursor techniques which provide disambiguation mechanisms for multiple intersected targets. The *Lock Ray* tool augments the ray cursor with a depth marker and performs the selection and disambiguation phases in a two-step process. First the user selects the ray. Once it is locked, the depth marker appears. Then, forward and backward hand movements fix the depth marker and the intersected target closest to it is highlighted in red indicating that it can be selected by releasing the button. Our approach also decouples the selection and disambiguation phases in two sequential steps, though using a cycling method for candidate selection [12]. In contrast to these previous works, we work directly with the captured volume dataset (using DVR) without any kind of costly preprocess to extract the isosurfaces.

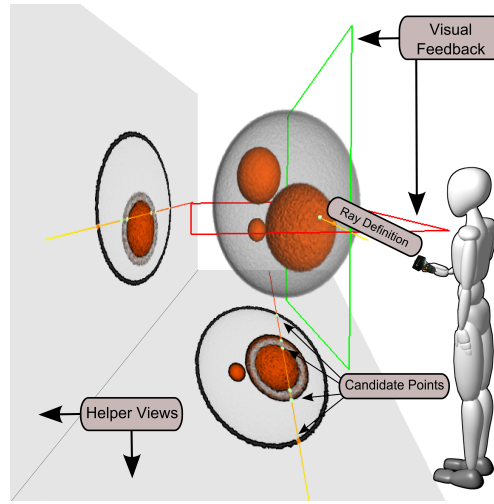
### 3 DAAPMed metaphor

Our objective is a user-friendly, efficient, accurate anchor point selection technique that facilitates getting measurements in VR environments with medical models. We also require ease of use and limiting the amount of effort the user has to perform. In contrast to normal desktop environment, where working with a mouse allows users to rest the arm, in 3D environments users usually do ample arm movements and have no surface to rest. With the objective of facilitating



the integration with the specialists' clinical work, we directly use the captured volume dataset rendered using DVR with a transfer function that shows semi-transparent and opaque structures (see Figures 1 and 2).

As a first approach, we extended and adapted to 3D the classical desktop point selection using a clipping plane (see Figure 3). However, as shown in our user study, this metaphor requires quite a long time and large movements from the user, because the correct definition of a clipping plane suitable for posterior point selection is difficult. In order to overcome these limitations we have developed a ray-based approach that uses the data information to facilitate the ray setting and point selection, reducing time and displacements.



**Fig. 2.** DAAPMed metaphor: A ray is used for selection, and a couple of supporting planes help the user to locate it in relation to the 3D structures. Potential anchor points, represented by colored small spheres, are computed as the intersections of the ray with the isosurfaces. Finally, Helper Views provide a better perception of the ray position as well as aids disoccluding interior candidate points. Notice that Helper Views show that the large orange sphere is hollow.

### 3.1 Data aware 3D selection Metaphor

Studies have shown that ray-based selection techniques often result in faster selection times than hand extension techniques in VR environments [13]. Unfortunately, as commented previously, ray cursor techniques have an inherent problem: the ray may intersect multiple objects. A naïve approach simply selects the first target which is intersected; however, it becomes very difficult or even impossible to select occluded points. Thus a more sophisticated method is required. The DAAPMed metaphor has three main components (see Figure 2):

- Ray cursor tool: It casts a pointing ray through the volume. The ray path visualization is enriched with the candidate selection points and its supporting planes, which provide a better insight of its position and orientation.
- Helper Views: We provide two views that help to understand the position of the ray inside the volume. This extra-visualization is inspired by the Magic Mirrors View [14], but, instead of showing the whole model, our view shows the model clipped by a plane that enables the possibility of showing the ray trajectory without any occlusion.
- Disambiguation mechanism: Once the ray is locked, we may select among the different intersections of the ray with the isosurfaces in the model. We adopt the same solution as Hinckley [12] cycling from one target to the next.

The key difference with previous selection methods is the ability to work with volumetric models by automatically generating candidate points through a rapid isosurface detection. Moreover, we also add visual cues that facilitate the understanding of the ray position and orientation, and disocclude inner intersection points.

Figure 2 shows all the components involved in the metaphor. In this example, the dataset consists of four spheres of different materials. The metaphor works as follows: when the user presses the back button of the input device, the *selection task* starts and the ray is painted with a gradient color from red to yellow (in this way we provide users with a visual cue of the depth of the ray). Throughout this process, the system continuously computes the proper set of candidate points. This set is composed by all the intersections of the ray with the implicitly defined isosurfaces. Upon button release, the last ray shown is locked, meaning that the selection phase has finished and the disambiguation task begins. The nearest candidate point is marked in orange (default selection) and the rest of the points are in white. The joystick provided by the input device allows the user to cycle between all the candidate points. This is convenient because it reduces movements. Since candidate points may have a random distribution, tracking the user’s movement to reach all the candidate points without a large arm movement (as proposed in [11]) would be difficult and might result in large varying patterns for different rays of the same volume.

As the 3D ray is painted over the volume, it is sometimes difficult to interpret how the volume is traversed. In order to give the user a second cue on the intersection of the ray with the volume, we provide the Helper Views. These showed to be of great utility, since some candidate points are usually occluded by other parts of the volume (Figure 1 shows a snapshot of our technique). We augment the visualization of the volume models with a wireframe representation of the cutting planes used in the Helper Views in order to provide the users with a visual feedback of the placement of such planes.

### 3.2 Implementation Details

In this section we give some details on how the isosurfaces are detected in real time as well as on how the Helper Views are created. One key difference with

other anchor point selection methods is the automatic detection of isosurfaces on-the-fly along the pointing ray. Since we have a non-segmented model, this isosurfaces must be determined in real-time, as they depend on the transfer function. Throughout all the process we use a DVR method using a GPU-based ray casting.

**Ray - Isosurface Intersection Detection** Volumetric models can be seen as a 3D scalar function  $f : V \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}$  (e.g. density value of a material). Let  $TF : \mathbb{R} \rightarrow \mathbb{R}^4$  be the transfer function used in the volume rendering algorithm, that assigns color and opacity to a scalar property. First of all, we have to define the conditions that a point  $p$  of the volume dataset  $V$  must fulfill to be considered a boundary-surface candidate point. These are:

1.  $p$  must belong to a visible material. This condition can be expressed formally as  $opacity(TF(f(p))) > 0.0$
2.  $p$  must belong to the boundary of a well-defined isosurface. This condition is satisfied if:
  - (a) The gradient at point  $p$ ,  $\nabla f(p)$ , has to be well defined. This means that  $\|\nabla f(p)\|$  is larger than a certain threshold. This threshold is automatically set by a previous analysis of the range of the magnitudes of the gradient.
  - (b) There exists a change in the sign of the direction of the gradient at  $p$  at the neighborhood of  $p$ . This property expresses the fact that the boundary passes through  $p$ .

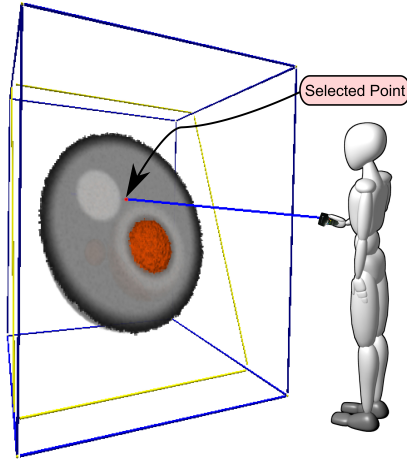
Since the detection of the boundary condition (2.b) may not be real-time in a VR environment, the information necessary to test this condition is precomputed. This is carried out by applying a 3D edge detection process [15] to the volume  $V$  and storing the result in a 3D texture which consists of a value per voxel that indicates if the voxel is being crossed by the boundary of a surface. Our system guarantees testing at least a point for each voxel intersected by the ray, thus the accuracy of our approach is related to voxel's size. As shown in Section 4, we obtain an accuracy comparable to the clipping plane selection approach. We also compared the accuracy with a desktop-based approach (using the raycasting paradigm for the nearest point selection) which works with a triangle mesh obtained using the Marching Cubes algorithm from the same volume dataset than in our VR-based metaphor. We obtained errors that did not differ significantly, which demonstrates that we may achieve comparable results in a VR environment. In both cases, the error performed was below the voxel size.

**Helper Views** The goal of Helper Views is to provide additional information on the exact position of the ray inside the volume. These views are drawn on two fixed planes, located to the left (YZ) and bottom (XZ) of the volume dataset (see Figure 2). Images displayed on these views are generated with the same algorithm used for rendering the volume dataset but clipping it by the plane that contains the ray and is the most parallel to each of the image planes YZ and XZ,

respectively. This has a main advantage: it shows the candidate points that lie inside the volume, therefore facilitating ray selection without previous manipulation of the volume (i. e. clipping). The visualization of the cut volume dataset is enhanced with an illustrative motif: view-dependent contours computed by using a Sobel kernel, highlighting the silhouette of the clipped region.

## 4 Evaluation and Results

We have conducted a formal user study to evaluate the accuracy, efficiency and ease of use of our approach. We take as a reference an implementation of the Clipping Plane (CP) selection method, since it is a technique that has been widely used in medical applications. We have found that the users required far less movement with our system than with CP, moreover DAAPMed technique was more accurate than the CP technique. We performed the user study in an immersive virtual reality setup composed of a  $2.7 \times 2$  meters passive stereo PowerWall. Users were tracked using an Intersense IS-900 Motion Tracking System device consisting on a Head Tracker and a MiniTrax Wanda with a joystick and five programmable buttons.



**Fig. 3.** Adaptation to VR of the clipping plane technique for selecting points located on it.

### 4.1 Design details of the Clipping Plane technique in Virtual Reality

In order to compare our selection technique with the classical approach for anchor point selection using clipping planes (CP), we ported this metaphor the following

way. Two buttons of the input device allows the user to set the action to be performed: rotate or translate the clipping plane. While the user is pressing the selected button, the clipping plane is rotated or translated accordingly to the user’s hand movement. Once the plane is fixed, the user can select a point on it using the ray-cursor paradigm. By pressing another button, the user indicates the desired point, so every point inside the volume, belonging to the plane, could be a candidate point to be selected. However, due to changes in the holding forces done by users when pressing or releasing a button, called Heisenberg effect [16], the accuracy of the selection may be affected. In order to solve this problem, we enhanced the visualization of the ray with a freezing timer.

## 4.2 Test design

Medical doctors often address two different point selection problems: selection of well-established anatomical points and distance measurement.

As a consequence, we decided to test two different tasks: the selection of individual points ( $T_1$ ), and the measurement of distances ( $T_2$ ). In order to get more information on the performance of our method, we added an extra experiment tailored to determine the accuracy of point selection ( $T_3$ ). The processes of each task were defined as:

- In  $T_1$  task, users had to introduce two anchor points ( $P_1$  and  $P_2$ ) at positions which were marked in the model with the use of a cone representation.
- To solve  $T_2$ , it was required to take a measure (calculated as a distance between two anchor points).
- $T_3$  consisted of locating, as accurately as possible, a set of points with the helping of a image shown in the bottom left corner of the screen.

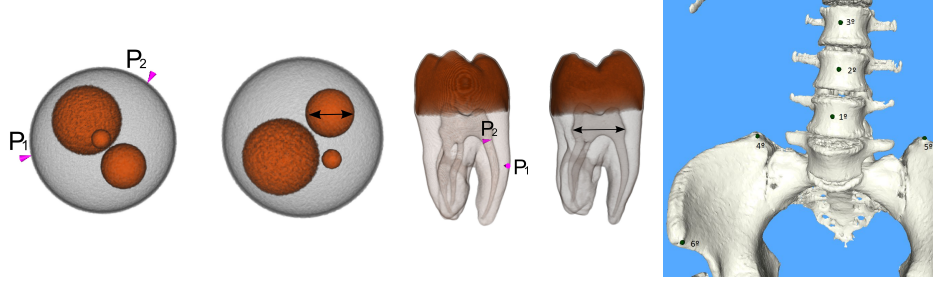
We performed two sessions: one consisted of  $T_1$  and  $T_2$  (we called this session as  $Test_1$ ) and the other called  $Test_2$  conformed by task  $T_3$ .

**Data preparation** Regarding  $Test_1$  we prepared two different datasets. The first one was used for training, while the other was used for the test. The training model consisted of a set of four spheres of different materials (Figure 4-left). The second model consisted of a typical CT dataset in volume visualization, a tooth, using a transfer function which shows the outside and inner shape of it (Figure 4-middle). The anchor points used in task  $T_1$  were chosen between external and internal characteristics of the model (see Figure 4).

The model used in  $Test_2$ , was taken from a medical dataset. Figure 4-right shows the helping image presented to users in task  $T_3$ .

We recorded the following indicators for each task:

- Task completion time (TCT): We measured the amount of time devoted to complete each task.
- Input Device Footprint (IDF): We measured the length of the total path followed by the device to complete each task.



**Fig. 4.** The datasets involved in the user study. The left images show the training dataset used in  $Test_1$ . The testing dataset in  $Test_1$  is shown in the center block images. Each figures block show the description of the tasks  $T_1$  and  $T_2$ . The right image shows the model used in  $T_3$  task and the set of points users have to select as accurately as possible.

- User footprint (UF): It measures the user displacement inside the VR environment done while carrying out the task.
- Accuracy (Ac): This value measures the error in the selection with respect to the reference points, taking into account the dimension of the voxel dataset as a metric of the error made. The model used in  $T_1$  and  $T_2$  task has a resolution of  $256 \times 256 \times 256$  and a voxel dimension of  $1.0 \times 1.0 \times 1.0$  mm. The model used in  $T_3$  task has a resolution of  $512 \times 512 \times 369$  and a voxel dimension of  $2.042 \times 2.042 \times 3.56$ mm.

**Subjects and Procedure.** 17 subjects participated in the evaluation; 13 male and 4 female, ranging between 23 and 63 years old. Subjects were asked to classify (as Low, Medium or High) their experience in a VR setup, experience with input devices and expertise in 3D application. All of the participants were people from our department: computer scientists at different levels of studies (master and PhD students) and faculty staff.

All the subjects participated in  $Test_1$ . In  $Test_2$ , only a subset of it participated (13 subjects: 10 male and 3 female, ranging between 23 and 40 years old). Every user performed each test once.

Before the first test started, a complete training (using the spheres dataset) was performed for the users to get familiar with the two interaction techniques to evaluate. Each test was divided into two blocks, one for each technique. The order of the blocks was chosen randomly in order to avoid skewing one of the techniques with a learning effect.

As said before,  $Test_1$  consisted of two kind of tasks: selecting two predefined points ( $T_1$ ), and measuring a certain distance ( $T_2$ ). For  $T_1$ , we asked the users to introduce two anchor points ( $P_1$  and  $P_2$ ) at positions that were marked in the model with the use of a cone representation (see Figure 4). Once completed, we stopped tracking the movements of the user until he or she was ready for the next task.  $T_2$  consisted of taking a measure (calculated as a distance between

two anchor points). The specification of this task was accompanied with different descriptions and pictures of the goal (we used the ones shown in Figure 4). None of the users involved in the experiment had any problem understanding the objective of the task. Users were allowed to repeat the selection of a point as many times as needed, until the point was validated.

In the *Test<sub>2</sub>* session, we proceed in the same way as the first one. Each participant performs the test once.

### 4.3 Statistical Results

A repeated measures within subjects design was used. The independent variable was the technique and the dependent variables were the set of tracked variables. A one-way analysis of variance (ANOVA) comparing both techniques was used.

Table 1 summarizes the statistical analysis of the relevant variables for test *Test<sub>1</sub>*. For each variable the mean and the standard deviation are shown. Task *T<sub>1</sub>* is tagged as *P<sub>1</sub>* and *P<sub>2</sub>*, corresponding to the two anchor points.

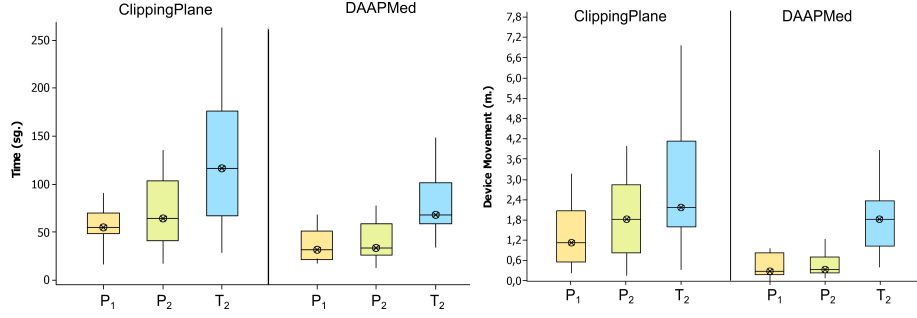
	CP			DAAPMed		
	<i>P<sub>1</sub></i>	<i>P<sub>2</sub></i>	<i>T<sub>2</sub></i>	<i>P<sub>1</sub></i>	<i>P<sub>2</sub></i>	<i>T<sub>2</sub></i>
Ac	0.76 ± 0.23	0.93 ± 1.37	1.15 ± 0.81	0.56 ± 0.23	1.37 ± 3.13	1.08 ± 0.79
TCT	62.42 ± 34.08	73.8 ± 47.1	119.8 ± 65.5	43.07 ± 36.5	41.1 ± 25.7	84.1 ± 43.4
IDF	3.711 ± 2.75	4.86 ± 4.8	7.92 ± 5.57	2.33 ± 3.26	1.88 ± 1.46	5.42 ± 3.31
UF	1.94 ± 1.53	2.41 ± 2.33	4.281 ± 3.23	1.33 ± 1.87	1.26 ± 1.04	2.83 ± 1.79

**Table 1.** The overall statistical results of the evaluation shown as means and standard deviations of the variables measured for test *Test<sub>1</sub>*. Regarding the mean and the standard deviation, DAAPMed is superior to CP. The one-way ANOVA analysis will show which differences were statistically significative.

Regarding *Completion Time*, there is significant evidence in all the experiments that DAAPMed performed better than CP. For *P<sub>1</sub>* ( $p = 0.028$ ,  $F = 5.83$ ), for *P<sub>2</sub>* ( $p = 0.008$ ,  $F = 9.35$ ) and for *T<sub>2</sub>* ( $p = 0.044$ ,  $F = 4.79$ ). Figure 5-left shows the total time for each technique.

Regarding *Input Device Footprint*, we measured the length of the total path which the device took to complete the experiment. We have found a significant effect on the *Input Device Footprint* variable for *P<sub>1</sub>* ( $p = 0.036$ ,  $F = 5.24$ ) and for *P<sub>2</sub>* ( $p = 0.004$ ,  $F = 11.70$ ). Figure 5-right illustrates the effect of the reduction of the footprint for DAAPMed technique. The reduction of footprint is especially important since a handheld 6-DOF device is being used, which can lead to fatigue with extended use [17].

We also measured the movement carried out by the user. In all cases, our system requires a lower amount of movement by the user. The analysis shows that the movement done in DAPPMed is significantly less than CP for *P<sub>2</sub>* ( $p = 0.009$ ,  $F = 8.72$ ) and for *T<sub>2</sub>* ( $p = 0.03$ ,  $F = 5.62$ ).



**Fig. 5.** Results of the completion task timings (left) and the Input Device footprint (right) for test  $Test_1$ . The boxes show the interquartile range with the median as the horizontal bar. The whiskers extend to the minimum and maximum of the data. CP exhibits longer selection times. Regarding the Input device footprints, it is clear that DAAPMed method performed significantly better than CP.

Table 2 summarizes the statistical analysis of the relevant variables for test  $Test_2$ . The first and second rows show the mean and the standard deviation for both techniques. The third and fourth rows show the statistical significance information ( $p$  and  $F$ ). For all the points introduced (except  $P_4$  and  $P_5$ ), the DAAPMed technique shows significant statistical difference with respect CP. Figure 6 shows a boxplot for all the performed tasks.

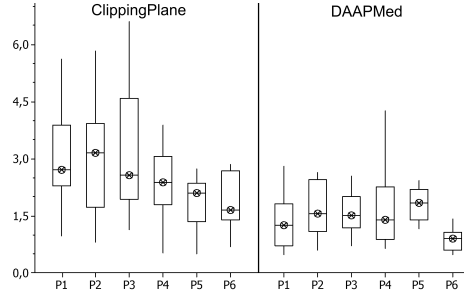
	$P_1$	$P_2$	$P_3$	$P_4$	$P_5$	$P_6$
CP	$2.94 \pm 1.30$	$3.02 \pm 1.49$	$3.17 \pm 1.73$	$2.34 \pm 0.88$	$2.07 \pm 1.09$	$2.07 \pm 1.17$
DAAPMed	$1.29 \pm 0.67$	$1.70 \pm 0.70$	$1.60 \pm 0.50$	$1.77 \pm 0.57$	$1.79 \pm 0.42$	$0.28 \pm 0.08$
$p, F$	0.002 - 16.55	0.011 - 9.01	0.005 - 11.58	0.19 - 1.96	0.38 - 0.81	0.001 - 17.42

**Table 2.** The overall statistical results of the evaluation for the test  $Test_2$  shown as means and standard deviations of the tolerance error. We can clearly see how the DAPPMed metaphor provides better results for all the tasks than the CP method.

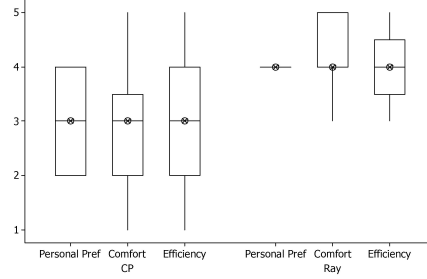
#### 4.4 Post-questionnaire results

To complete the information, we also asked the subjects to fill some questionnaires, to know the preferences of the users between the two techniques. All responses in the post-questionnaire were measured on a Likert scale of 1-5, where 1 meant the worst value and 5 was the best value. The results are shown in Figure 7. The answers seem to indicate that DAAPMed metaphor is more suitable than the CP technique.





**Fig. 6.** Accuracy by technique. The boxes show the interquartile range with the median as the horizontal bar. The whiskers extend to the minimum and maximum of the data.



**Fig. 7.** Results obtained from a personal preference evaluation questionnaire. These results show that the users' perceptions are quite positive with our tool.

The users noted two major problems with respect to our technique. The first one is the inherent jittering of the tracker, that made selection affect user performance. Only two users agreed in that it seems to produce a more relevant effect to the ray-based selection than to the plane-based. Furthermore, in all the experiments, the ray-based approach showed a better behavior than the clipping-planes system. The second issue was the lack of ray refinement: most users suggested that a fine tuning of the ray, after its initial positioning would be welcome. We let this work for future improvements.

## 5 Picking optimizations

Working with 3D input devices require a steady hand in order to obtain an accurate selection due to the inherent jittering of the tracker. However, due to changes in the holding forces done by users when pressing or releasing a button, called Heisenberg effect [16], the accuracy of the selection may be affected. In order to solve this problem, we enhanced the visualization of the ray with a freezing timer. Though providing better feedback to the user, there are still some

users that shake their hands inadvertently and this often affects the selection accuracy.

### 5.1 Shake filtering

In order to reduce the effect of quivery hands, we combine the use of the freezing time with an averaging of the captured position. Trembling hands affect the overall performance and produce a bad sensation of using this kind of input devices and on account of this a complete refusal of the use of virtual reality. Our algorithm filters the selection position by taking the average of the last 20 captured positions by the tracker. Moreover, we further check whether the final position falls within a maximum tolerance range from the position at which the selection button was initially pressed.

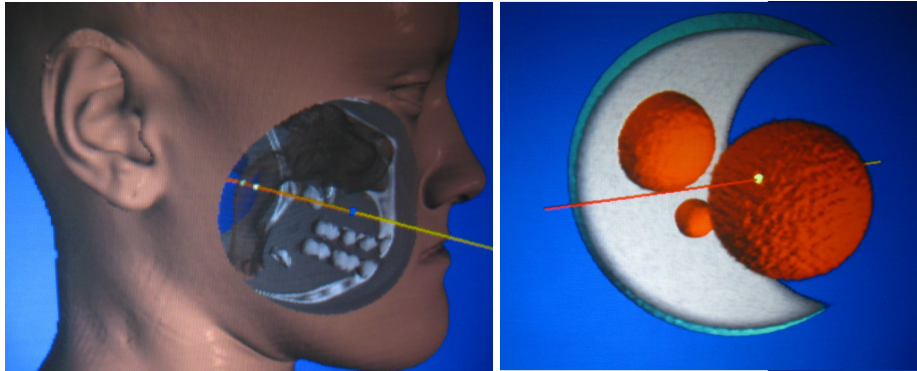
Although not a full user study has been performed, we took two of the users that showed a bad steady hand, and they experimented with the improved method. In one case, the results were similar (with a precision improvement of around the 10%), but the second showed an increase in precision of around the 40%. These results look promising, but further tests have to be carried out.

### 5.2 Extending the selection candidates using VML

In Volume Rendering, it is difficult to simultaneously visualize interior and exterior structures because the structures are commonly quite complex and it is easy to lose the context. The *Virtual Magic Lantern* [3] (VML) is a specialized interaction tool tailored to facilitate the inspection of a volume dataset in VR environments. It addresses the occlusion management problem, facilitating the inspection of inner structures without the total elimination of the exterior structures, offering in this way, a context-based visualization of the overall structures. We have made more powerful the DAAPMed technique increasing its capabilities coupling it with the VML metaphor. In this sense, we build the VML through the use of the 3D pointer device that casts a cylinder onto the model. The axis of the cylinder is defined from the orientation of the 3D pointer device. The volume not intersected by the cylinder is rendered using the original transfer function while the volume intersected by it is rendered using the second transfer function (see Figure 8). Regarding DAAPMed, the *Ray cursor tool* fit in the axis of the cylinder. The set of candidate selection points are calculated taking into account the transfer function used inside the cylinder. In this way, the user can pick points inside a volume dataset without losing the overall context provided by the other transfer function.

## 6 Conclusions

We have presented a new interaction technique for picking points in a volume dataset. This selection technique follows the *ray casting* paradigm, enhanced with an automatic calculation of the set of suitable points of interest by an



**Fig. 8.** These images illustrate the integration of both techniques. The left image shows a head where the outside of the VML consists of the skin, while its interior shows the air-cavities in the body. The right image shows one of the models from the user study. The most exterior sphere is visualized with a blue opaque material, while the transfer function inside the VML is the same transfer function that was used in the user study.

on-the-fly determination of the isosurfaces along the ray path. The feedback with the interaction is enhanced with a meaningful visualization called *Helper Views* that provides context for the ray selection and shows occluded detected candidate points that would be otherwise invisible to the user without posterior and ad-hoc volume manipulation.

The user study showed that our technique is easy to learn and to use. Despite the limited precision of the 3D input devices, our technique achieves a precise 3D interaction thanks to the automatic anchor point calculation provided by the system. Users felt more comfortable and achieved better results with our system than with the clipping plane technique.

## 7 Acknowledgments

The authors want to thank all the participants involved in the user study. This work has been supported by the project TIN2010-20590-C01-01 of the Spanish Government.

## References

1. Reitinger, B., Schmalstieg, D., Bornik, A., Beichel, R.: Spatial analysis tools for virtual reality-based surgical planning. *3D User Interfaces* **0** (2006) 37–44

2. Monclús, E., Vázquez, P.P., Navazo, I.: DAAPMed: A data-aware anchor point selection tool for medical models in vr environments. In: VISIGRAPP - International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. (2013) 308–317
3. Monclús, E., Díaz, J., Navazo, I., Vázquez, P.P.: The virtual magic lantern: an interaction metaphor for enhanced medical data inspection. In: VRST '09: Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology, New York, NY, USA, ACM (2009) 119–122
4. Hinckley, K., Pausch, R., Goble, J.: A three-dimensional user interface for neurosurgical visualization. In: In the SPIE Conf. on Medical Imaging, SPIE (1994) 126–136
5. Preim, B., Tietjen, C., Spindler, W., Peitgen, H.O.: Integration of measurement tools in medical 3d visualizations. In: Visualization '02, IEEE Computer Society (2002) 21–28
6. Rössling, I., Cyrus, C., Dornheim, L., Boehm, A., Preim, B.: Fast and flexible distance measures for treatment planning. *International Journal of Computer Assisted Radiology and Surgery* **5** (2010) 633–646
7. Hagedorn, J., Joy, P., Dunkers, S., Peskin, A.: Measurement tools for the immersive visualization environment: Steps toward the virtual laboratory. *Journal of Research of the National Institute of Standards and Technology* **112**(5) (2007) 257–270
8. Hastreiter, P., Tom, B., Eberhardt, K.E.W., Ertl, T.: Fast analysis of intracranial aneurysms based on interactive direct volume rendering and ct angiography. In: CTA, Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer (1998) 660–669
9. Gallo, L., De Pietro, G., Marra, I.: 3d interaction with volumetric medical data: experiencing the wiimote. In: Proceedings of the 1st international conference on Ambient media and systems. (2008) 14:1–14:6
10. Bowman, D., Kruijff, E., LaViola, J., Poupyrev, I.: 3D User Interfaces: Theory and Practice. Addison-Wesley, Pearson Education (2004)
11. Grossman, T., Balakrishnan, R.: The design and evaluation of selection techniques for 3d volumetric displays. In: Proceedings of the Symposium on User interface software and technology, ACM (2006) 3–12
12. Hinckley, K., Pausch, R., Goble, J., Kassell, N.: A survey of design issues in spatial input. In: Proceedings of the Symposium on User interface software and technology, ACM (1994) 213–222
13. Bowman, D.A., Johnson, D.B., Hodges, L.F.: Testbed evaluation of virtual environment interaction techniques. In: Proceedings of the ACM symposium on Virtual reality software and technology. (1999) 26–33
14. König, A.H., Doleisch, H., Gröller, E., Brain, T.H.: Multiple views and magic mirrors - fmri visualization of the human brain (1999)
15. Monga, O., Deriche, R., Malandain, G., Cocquerez, J.P.: Recursive filtering and edge closing: two primary tools for 3-d edge detection. In: Proceedings of the first european conference on Computer vision, Springer-Verlag (1990) 56–65
16. Bowman, D.A., Wingrave, C.A., Campbell, J.M., Ly, V.Q., Rhoton, C.J.: Novel uses of pinch gloves for virtual environment interaction techniques. *Virtual Reality* **6** (2002) 122–129
17. Ware, C., Slipp, L.: Using velocity control to navigate 3d graphical environments: A comparison of three interfaces. In: Human Factors and Ergonomic Studies (HFES) Meeting. (1991) 25–32